Notice of Allowability	Application No.	Applicant(s)	
	09/912,389	COWIE ET AL.	
	Examiner	Art Unit	
	MATTHEW T. HENNING	2131	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *the appeal brief filed 3/4/2008*.

2. ☒ The allowed claim(s) is/are *1-3,5-8,12,15-19,21-24,28,31-35,37-40,44,47-51,53-56,60,63-67,69-72,76,79-83,85-88,92 and 95-98*.

3. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☐ All b) ☐ Some* c) ☐ None of the:
      1. ☐ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
      3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
      1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
   (b) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

   Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☒ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☐ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date _____
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application
6. ☒ Interview Summary (PTO-413), Paper No./Mail Date *20080519* .
7. ☒ Examiner's Amendment/Comment
8. ☒ Examiner's Statement of Reasons for Allowance
9. ☐ Other _____.

EXAMINER'S AMENDMENT

An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Kevin Zilka on 5/19/2008.

The application has been amended as follows:

Please replace all previous claims with the claim listing which begins on the following page:

1. (Currently Amended) A computer program product in a computer storage medium comprising a computer program operable to control a computer to detect a known computer program within a packed computer file, said packed computer file being unpacked upon execution, said computer program comprising:

resource data reading logic for reading resource data within said packed computer file, said resource data specifying program resource items used by said known computer program and readable by a computer operating system without dependence upon which unpacking algorithm is used by said packed computer file; and

resource data comparing logic for generating characteristics of said resource data and for comparing said characteristics of said resource data with characteristics of known computer program resource data and for detecting a match with said known computer program indicative of said packed computer file containing said known computer program;

wherein said resource data of said packed computer file is processed to generate fingerprint data and to compare said generated fingerprint data with known computer program fingerprint data;

wherein said generated fingerprint data includes a number of program resource items specified within said resource data of said packed computer file;

wherein said generated fingerprint data includes a flag indicating which data is included within said generated fingerprint data;

wherein said generated fingerprint data includes a location within said resource data of said packed computer file of an entry specifying a program resource item having a largest size;

wherein said generated fingerprint data includes a checksum value calculated in dependence upon:

a number of said program resource items specified beneath each node within hierarchically arranged resource data of said packed computer file;

string names associated with said program resource items within said resource data of said packed computer file; and

sizes of said program resource items within said resource data of said packed computer file;

wherein said checksum value is rotated between each item being added into said checksum.

2. (Original) A computer program product as claimed in claim 1, wherein said known computer program is one of:

a Trojan computer program; and

a worm computer program.

3. (Previously Presented) A computer program product as claimed in claim 1, wherein said resource data comparing logic is operable to compare said resource data of said packed computer file with characteristics of a plurality of known computer programs to detect if said packed computer file contains one of said plurality of known computer programs.

4. (Cancelled)

5. (Original) A computer program product as claimed in claim 1, wherein said program resource items used by said known computer program include one or more of:

icon data;

string data;

dialog data;

bitmap data;

menu data; and

language data.

6. (Previously Presented) A computer program product as claimed in claim 1, wherein said resource data of said packed computer file specifies for each resource item a storage location of said resource item.

7. (Original) A computer program product as claimed in claim 6, wherein said storage location of said resource item is specified as an relative offset value.

8. (Previously Presented) A computer program product as claimed in claim 1, wherein said resource data of said packed computer file specifies for each resource item a size of said resource item.

9.-11. (Cancelled)

12. (Previously Presented) A computer program product as claimed in claim 1, wherein said generated fingerprint data includes timestamp data indicative of a time of compilation of said known computer program.

13. (Cancelled)

14. (Cancelled)

15. (Original) A computer program product as claimed in claim 1, wherein said packed computer file includes an unpacking computer program which upon execution decompresses said known computer program.

16. (Original) A computer program product as claimed in claim 1, wherein said packed computer file is a Win32 PE file.

17. (Currently Amended) A computer program product in a computer storage medium comprising a computer program operable to control a computer to generate data for detecting a known computer program within a packed computer file, said packed computer file being unpacked upon execution, said computer program comprising:

resource data reading logic for reading resource data within said packed computer file, said resource data specifying program resource items used by said known computer

program and readable by a computer operating system without dependence upon which unpacking algorithm is used by said packed computer file; and

characteristic data generating logic for generating characteristic data associated with said resource data for comparison with characteristic data of known computer program resource data to detect a match with said known computer program indicative of said packed computer file containing said known computer program;

wherein said resource data of said packed computer file is processed to generate fingerprint data and to compare said generated fingerprint data with known computer program fingerprint data;

wherein said generated fingerprint data includes a number of program resource items specified within said resource data of said packed computer file;

wherein said generated fingerprint data includes a flag indicating which data is included within said generated fingerprint data;

wherein said generated fingerprint data includes a location within said resource data of said packed computer file of an entry specifying a program resource item having a largest size;

wherein said generated fingerprint data includes a checksum value calculated in dependence upon:

a number of program resource items specified beneath each node within hierarchically arranged resource data of said packed computer file;

string names associated with program resource items within said resource data of said packed computer file; and

sizes of program resource items within said resource data of said packed computer file;

wherein said checksum value is rotated between each item being added into said checksum.

18. (Original) A computer program product as claimed in claim 17, wherein said known computer program is one of:

a Trojan computer program; and

a worm computer program.

19. (Previously Presented) A computer program product as claimed in claim 17, wherein said characteristic data generating logic is operable to generate characteristic data from a plurality of known computer programs to enable detection of any of said plurality of known computer programs within said packed computer file.

20. (Cancelled)

21. (Original) A computer program product as claimed in claim 17, wherein said program resource items used by said known computer program include one or more of:

icon data;
string data;
dialog data;
bitmap data;
menu data; and
language data.

22. (Previously Presented) A computer program product as claimed in claim 17, wherein said resource data of said packed computer file specifies for each resource item a storage location of said resource item.

23. (Original) A computer program product as claimed in claim 22, wherein said storage location of said resource item is specified as an relative offset value.

24. (Previously Presented) A computer program product as claimed in claim 17, wherein said resource data of said packed computer file specifies for each resource item a size of said resource item.

25.-27. (Cancelled)

28. (Previously Presented) A computer program product as claimed in claim 17, wherein said generated fingerprint data includes timestamp data indicative of a time of compilation of said known computer program.

29. (Cancelled)

30. (Cancelled)

31. (Original) A computer program product as claimed in claim 17, wherein said packed computer file includes an unpacking computer program which upon execution decompresses said known computer program.

32. (Original) A computer program product as claimed in claim 17, wherein said packed computer file is a Win32 PE file.

33. (Currently Amended) A method of controlling a computer to detect a known computer program within a packed computer file, said packed computer file being unpacked upon execution, said method comprising the steps of:

reading resource data within said packed computer file, said resource data specifying program resource items used by said known computer program and readable by a computer operating system without dependence upon which unpacking algorithm is used by said packed computer file; and

generating characteristics of said resource data and comparing said characteristics of said resource data with characteristics of known computer program resource data and detecting a match with characteristics of said known computer program indicative of said packed computer file containing said known computer program;

wherein said resource data of said packed computer file is processed to generate fingerprint data and to compare said generated fingerprint data with known computer program fingerprint data;

wherein said generated fingerprint data includes a number of program resource items specified within said resource data of said packed computer file,

wherein said generated fingerprint data includes a flag indicating which data is included within said generated fingerprint data;

wherein said generated fingerprint data includes a location within said resource data of said packed computer file of an entry specifying a program resource item having a largest size;

wherein said generated fingerprint data includes a checksum value calculated in dependence upon:

a number of program resource items specified beneath each node within hierarchically arranged resource data of said packed computer file;

string names associated with program resource items within said resource data of said packed computer file; and

sizes of program resource items within said resource data of said packed computer file;

wherein said checksum value is rotated between each item being added into said checksum.

34. (Original) A method as claimed in claim 33, wherein said known computer program is one of:

a Trojan computer program; and

a worm computer program.

35. (Previously Presented) A method as claimed in claim 33, wherein said step of comparing compares said resource data of said packed computer file with characteristics of a plurality of known computer programs to detect if said packed computer file contains one of said plurality of known computer programs.

36. (Cancelled)

37. (Original) A method as claimed in claim 33, wherein said program resource items used by said known computer program include one or more of:

icon data;

string data;
dialog data;
bitmap data;
menu data; and
language data.

38. (Previously Presented) A method as claimed in claim 33, wherein said resource data of said packed computer file specifies for each resource item a storage location of said resource item.

39. (Original) A method as claimed in claim 38, wherein said storage location of said resource item is specified as an relative offset value.

40. (Previously Presented) A method as claimed in claim 33, wherein said resource data of said packed computer file specifies for each resource item a size of said resource item.

41.-43. (Cancelled)

44. (Previously Presented) A method as claimed in claim 33, wherein said generated fingerprint data includes timestamp data indicative of a time of compilation of said known computer program.

45. (Cancelled)

46. (Cancelled)

47. (Original) A method as claimed in claim 33, wherein said packed computer file includes an unpacking computer program which upon execution decompresses said known computer program.

48. (Original) A method as claimed in claim 33, wherein said packed computer file is a Win32 PE file.

49. (Currently Amended) A method of controlling a computer to generate data for detecting a known computer program within a packed computer file, said packed computer file being unpacked upon execution, said method comprising the steps of:

reading resource data within said packed computer file, said resource data specifying program resource items used by said known computer program and readable by a computer operating system without dependence upon which unpacking algorithm is used by said packed computer file; and

generating characteristic data associated with said resource data for comparison with characteristic data of known computer program resource data and detecting a match with said known computer program indicative of said packed computer file containing said known computer program;

wherein said resource data of said packed computer file is processed to generate fingerprint data and to compare said generated fingerprint data with known computer program fingerprint data;

wherein said generated fingerprint data includes a number of program resource items specified within said resource data of said packed computer file;

wherein said generated fingerprint data includes a flag indicating which data is included within said generated fingerprint data;

wherein said generated fingerprint data includes a location within said resource data of said packed computer file of an entry specifying a program resource item having a largest size;

wherein said generated fingerprint data includes a checksum value calculated in dependence upon:

a number of program resource items specified beneath each node within hierarchically arranged resource data of said packed computer file;

string names associated with program resource items within said resource data of said packed computer file; and

sizes of program resource items within said resource data of said packed computer file;

wherein said checksum value is rotated between each item being added into said checksum.

50. (Original) A method as claimed in claim 49, wherein said known computer program is one of:

a Trojan computer program; and

a worm computer program.

51. (Previously Presented) A method as claimed in claim 49, wherein said step of generating generates characteristic data from a plurality of known computer programs to enable detection of any of said plurality of known computer programs within said packed computer file.

52. (Cancelled)

53. (Original) A method as claimed in claim 49, wherein said program resource items used by said known computer program include one or more of:

icon data;

string data;

dialog data;

bitmap data;

menu data; and

language data.

54. (Previously Presented) A method as claimed in claim 49, wherein said resource data of said packed computer file specifies for each resource item a storage location of said resource item.

55. (Original) A method as claimed in claim 54, wherein said storage location of said resource item is specified as an relative offset value.

56. (Previously Presented) A method as claimed in claim 49, wherein said resource data of said packed computer file specifies for each resource item a size of said resource item.

57.-59. (Cancelled)

60. (Previously Presented) A method as claimed in claim 49, wherein said generated fingerprint data includes timestamp data indicative of a time of compilation of said known computer program.

61. (Cancelled)

62. (Cancelled)

63. (Original) A method as claimed in claim 49, wherein said packed computer file includes an unpacking computer program which upon execution decompresses said known computer program.

64. (Original) A method as claimed in claim 49, wherein said packed computer file is a Win32 PE file.

65. (Currently Amended) Apparatus for detecting a known computer program within a packed computer file, said packed computer file being unpacked upon execution, said apparatus comprising:

a resource data reader for reading resource data within said packed computer file, said resource data specifying program resource items used by said known computer program and readable by a computer operating system without dependence upon which unpacking algorithm is used by said packed computer file; and

a resource data comparator for generating characteristics of said resource data and for comparing said characteristics of said resource data with characteristics of known computer program resource data for detecting a match with said known computer program indicative of said packed computer file containing said known computer program;

wherein said resource data of said packed computer file is processed to generate fingerprint data and to compare said generated fingerprint data with known computer program fingerprint data;

wherein said generated fingerprint data includes a number of program resource items specified within said resource data of said packed computer file;

wherein said generated fingerprint data includes a flag indicating which data is included within said generated fingerprint data;

wherein said generated fingerprint data includes a location within said resource data of said packed computer file of an entry specifying a program resource item having a largest size;

wherein said generated fingerprint data includes a checksum value calculated in dependence upon:

a number of program resource items specified beneath each node within hierarchically arranged resource data of said packed computer file;

string names associated with program resource items within said resource data of said packed computer file; and

sizes of program resource items within said resource data of said packed computer file;

wherein said checksum value is rotated between each item being added into said checksum.

66. (Original) Apparatus as claimed in claim 65, wherein said known computer program is one of:

a Trojan computer program; and

a worm computer program.

67. (Previously Presented) Apparatus as claimed in claim 65, wherein said resource data comparator is operable to compare said resource data of said packed computer file with characteristics of a plurality of known computer programs to detect if said packed computer file contains one of said plurality of known computer programs.

68. (Cancelled)

69. (Original) Apparatus as claimed in claim 65, wherein said program resource items used by said known computer program include one or more of:

icon data;
string data;
dialog data;
bitmap data;
menu data; and
language data.

70. (Previously Presented) Apparatus as claimed in claim 65, wherein said resource data of said packed computer file specifies for each resource item a storage location of said resource item.

71. (Original) Apparatus as claimed in claim 70, wherein said storage location of said resource item is specified as an relative offset value.

72. (Previously Presented) Apparatus as claimed in claim 65, wherein said resource data of said packed computer file specifies for each resource item a size of said resource item.

73.-75. (Cancelled)

76. (Previously Presented) Apparatus as claimed in claim 65, wherein said generated fingerprint data includes timestamp data indicative of a time of compilation of said known computer program.

77. (Cancelled)

78. (Cancelled)

79. (Original) Apparatus as claimed in claim 65, wherein said packed computer file includes an unpacking computer program which upon execution decompresses said known computer program.

80. (Original) Apparatus as claimed in claim 65, wherein said packed computer file is a Win32 PE file.

81. (Currently Amended) Apparatus for generating data for detecting a known computer program within a packed computer file, said packed computer file being unpacked upon execution, said apparatus comprising:

a resource data reader for reading resource data within said packed computer file, said resource data specifying program resource items used by said known computer program and readable by a computer operating system without dependence upon which unpacking algorithm is used by said packed computer file; and

a characteristic data generator for generating characteristic data associated with said resource data for comparison with characteristic data of known computer program resource data and for detecting a match with said known computer program indicative of said packed computer file containing said known computer program;

wherein said resource data of said packed computer file is processed to generate fingerprint data and to compare said generated fingerprint data with known computer program fingerprint data;

wherein said generated fingerprint data includes a number of program resource items specified within said resource data of said packed computer file;

wherein said generated fingerprint data includes a flag indicating which data is included within said generated fingerprint data;

wherein said generated fingerprint data includes a location within said resource data of said packed computer file of an entry specifying a program resource item having a largest size;

wherein said generated fingerprint data includes a checksum value calculated in dependence upon:

a number of program resource items specified beneath each node within hierarchically arranged resource data of said packed computer file;

string names associated with program resource items within said resource data of said packed computer file; and

sizes of program resource items within said resource data of said packed computer file;

wherein said checksum value is rotated between each item being added into said checksum.

82. (Original) Apparatus as claimed in claim 81, wherein said known computer program is one of:

a Trojan computer program; and

a worm computer program.

83. (Previously Presented) Apparatus as claimed in claim 81, wherein said characteristic data generator is operable to generate characteristic data from a plurality of known computer programs to enable detection of any of said plurality of known computer programs within said packed computer file.

84. (Cancelled)

85. (Original) Apparatus as claimed in claim 81, wherein said program resource items used by said known computer program include one or more of:

icon data;

string data;
dialog data;
bitmap data;
menu data; and
language data.

86. (Previously Presented) Apparatus as claimed in claim 81, wherein said resource data of said packed computer file specifies for each resource item a storage location of said resource item.

87. (Original) Apparatus as claimed in claim 86, wherein said storage location of said resource item is specified as an relative offset value.

88. (Currently Amended) Apparatus as claimed in claim 81, wherein said resource data of said packed computer file specifies for each resource item a storage location of said resource item

89.-91. (Cancelled)

92. (Previously Presented) Apparatus as claimed in claim 81, wherein said generated fingerprint data includes timestamp data indicative of a time of compilation of said known computer program.

93. (Cancelled)

94. (Cancelled)

95. (Original) Apparatus as claimed in claim 81, wherein said packed computer file includes an unpacking computer program which upon execution decompresses said known computer program.

96. (Original) Apparatus as claimed in claim 81, wherein said packed computer file is a Win32 PE file.

97. (Cancelled)

98. (Currently Amended) A computer program product as claimed in claim [[14]]1, wherein said checksum value is rotated 1 bit to the left.

Allowable Subject Matter

The following is an examiner's statement of reasons for allowance: The prior art does not provide teaching or suggestion of the inventive features in the combination as claimed, including generation of fingerprint data for packed computer files, wherein the fingerprint data includes a number of resource items specified within the resource data, a flag indicating which data is included within the fingerprint data, a location within said resource data of said packed computer file of an entry specifying a program resource item having a largest size, and a checksum value calculated in dependence upon a number of program resource items specified beneath each node within hierarchically arranged resource data, string names associated with the program resource items, and sizes of the program resource items, wherein the checksum value is rotated between each item being added into said checksum. While many of these claim features were, by themselves, common in the art at the time of invention, there is no teaching or suggestion in the prior to combine these features in the specific manner as claimed.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to MATTHEW T. HENNING whose telephone number is (571)272-3790. The examiner can normally be reached on M-F 8-4.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ayaz Sheikh can be reached on (571) 272-3795. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Matthew T Henning/
Art Unit 2131
/Ayaz R. Sheikh/

Supervisory Patent Examiner, Art Unit 2131